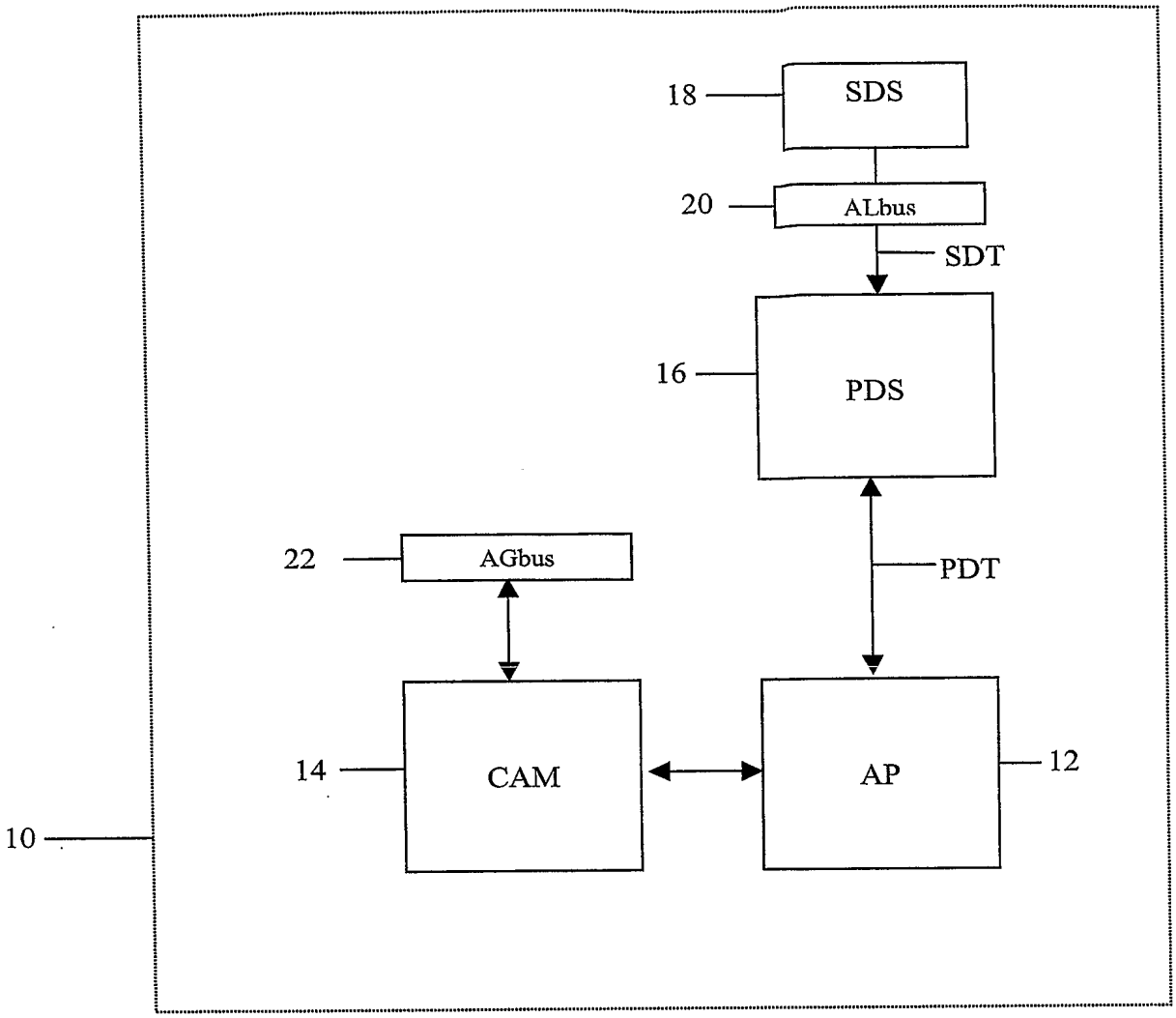


Fig. 1

Fig. 2

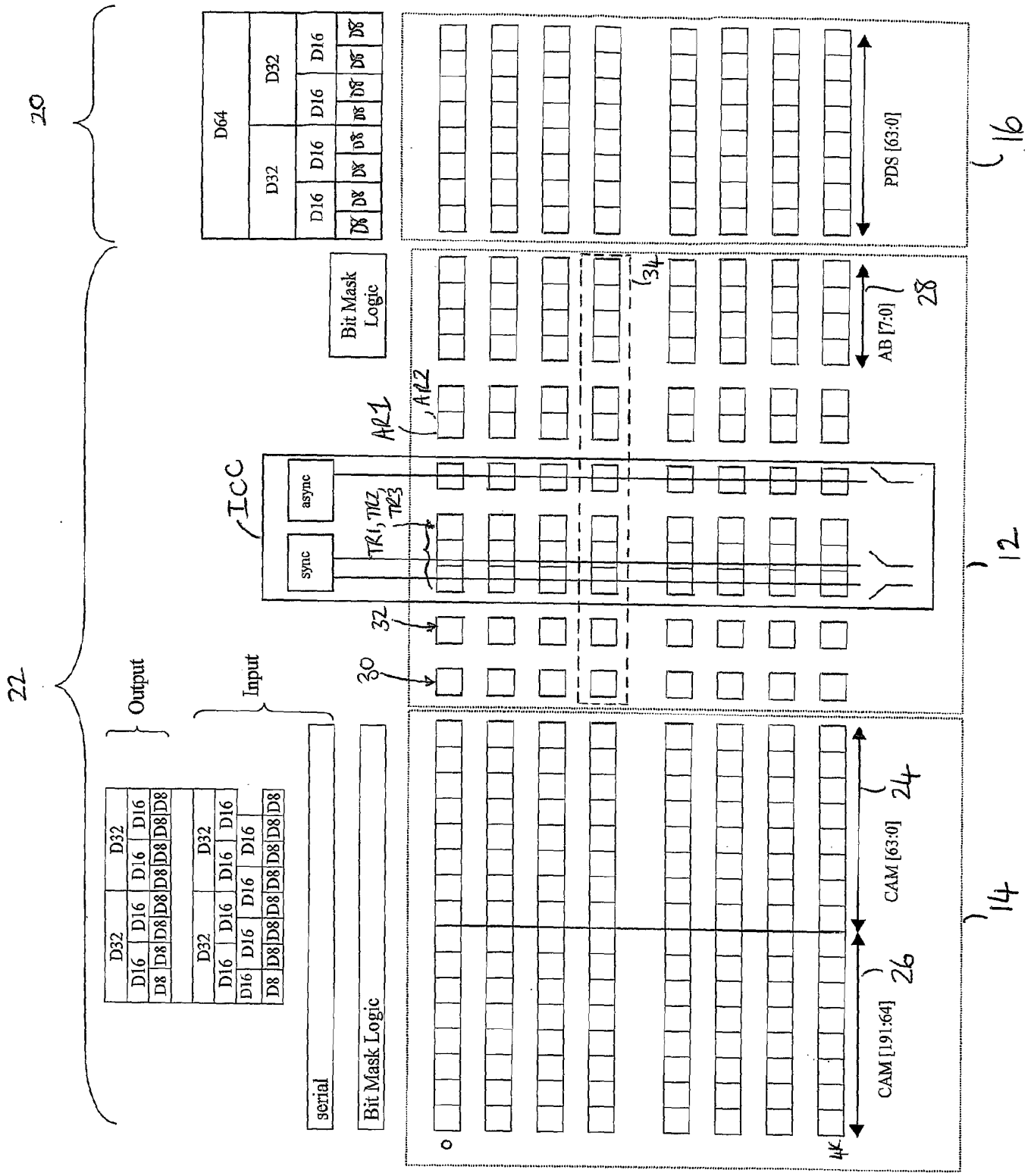
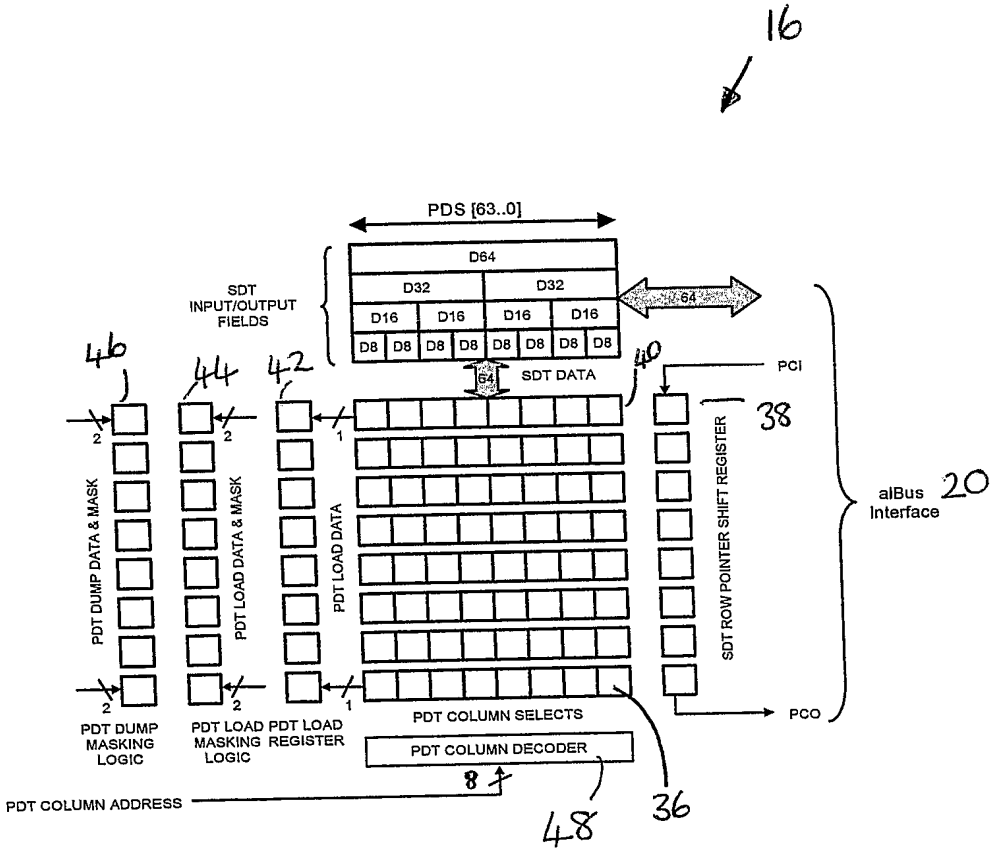

20
22
Output
Input
D32
D16
D8
D64
serial
Bit Mask Logic
30
32
ICC
sync
async
TR1, TR2, TR3
AR1
AR2
34
28
AB [7:0]
PDS [63:0]
16
12
14
24
26
CAM [63:0]
CAM [191:64]
0
4K

Fig. 3


16
PDS [63..0]
D64
D32
D32
D16
D16
D16
D16
D8
D8
D8
D8
D8
D8
D8
D8
64
SDT INPUT/OUTPUT FIELDS
64
SDT DATA
46
44
42
40
PCI
38
2
2
1
PDT DUMP DATA & MASK
PDT LOAD DATA & MASK
PDT LOAD DATA
SDT ROW POINTER SHIFT REGISTER
alBus Interface
20
PDT DUMP MASKING LOGIC
PDT LOAD MASKING LOGIC
PDT LOAD REGISTER
PDT COLUMN SELECTS
PCO
PDT COLUMN DECODER
8
PDT COLUMN ADDRESS
48
36

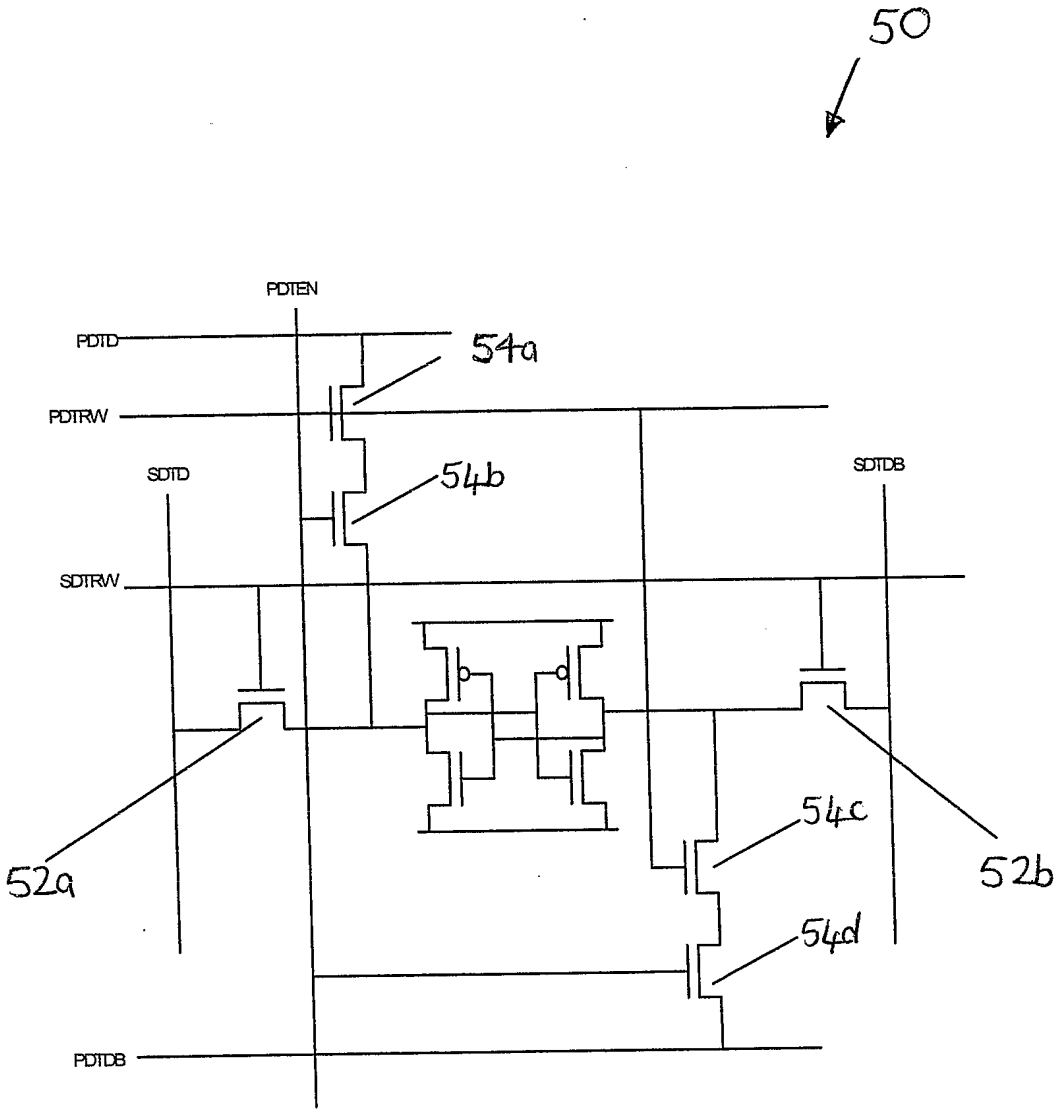

50
PDTEN
PDTD
54a
PDTRW
SDTD
54b
SDTDB
SDTRW
52a
54c
52b
54d
PDTDB

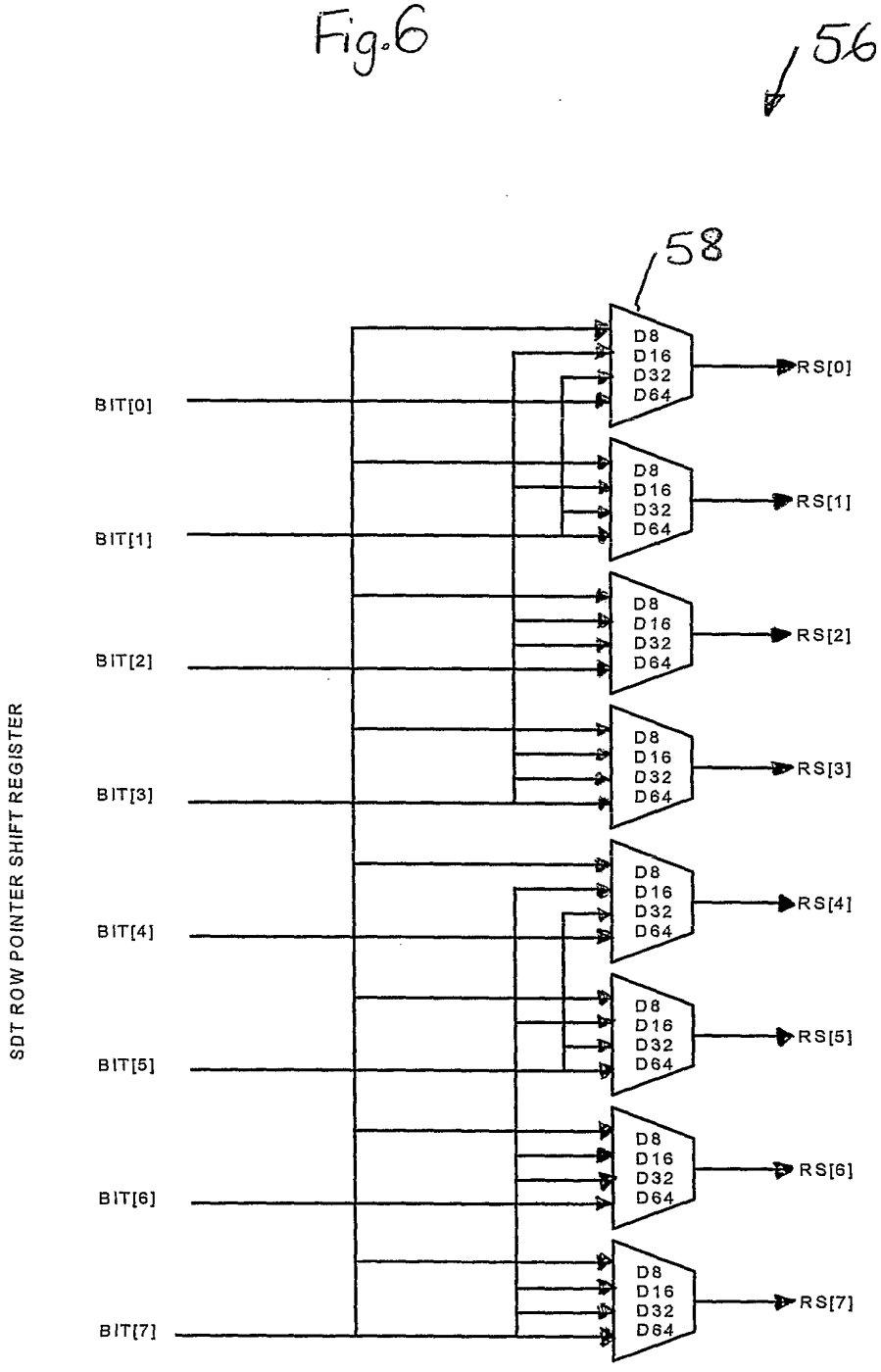
Fig. 4

row address MOD 8	Row Strobe logic conditions
0	SDTRW_A[0] = RS[0]
	SDTRW_B[0] = ~D8.RS[0]
	SDTRW_C[0] = (D32+D64).RS[0]
	SDTRW_D[0] = D64.RS[0]
1	SDTRW_A[1] = (D8+D64).RS[1]
	SDTRW_B[1] = (D16+D64).RS[1]
	SDTRW_C[1] = (D32+D64).RS[1]
	SDTRW_D[1] = D64.RS[1]
2	SDTRW_A[2] = ~D16.RS[2]
	SDTRW_B[2] = (D16+D64).RS[2]
	SDTRW_C[2] = (D32+D64).RS[2]
	SDTRW_D[2] = D64.RS[2]
3	SDTRW_A[3] = (D8+D64).RS[3]
	SDTRW_B[3] = ~D8.RS[3]
	SDTRW_C[3] = (D32+D64).RS[3]
	SDTRW_D[3] = D64.RS[3]
4	SDTRW_A[4] = (D8+D64).RS[4]
	SDTRW_B[4] = ~D8.RS[4]
	SDTRW_C[4] = (D32+D64).RS[4]
	SDTRW_D[4] = D64.RS[4]
5	SDTRW_A[5] = ~D16.RS[5]
	SDTRW_B[5] = (D16+D64).RS[5]
	SDTRW_C[5] = (D32+D64).RS[5]
	SDTRW_D[5] = D64.RS[5]
6	SDTRW_A[6] = (D8+D64).RS[6]
	SDTRW_B[6] = (D16+D64).RS[6]
	SDTRW_C[6] = (D32+D64).RS[6]
	SDTRW_D[6] = D64.RS[6]
7	SDTRW_A[7] = RS[7]
	SDTRW_B[7] = ~D8.RS[7]
	SDTRW_C[7] = (D32+D64).RS[7]
	SDTRW_D[7] = D64.RS[7]

Table 2

Fig. 5

Fig.6


56
58
SDT ROW POINTER SHIFT REGISTER
BIT[0]
BIT[1]
BIT[2]
BIT[3]
BIT[4]
BIT[5]
BIT[6]
BIT[7]
D8
D16
D32
D64
RS[0]
RS[1]
RS[2]
RS[3]
RS[4]
RS[5]
RS[6]
RS[7]

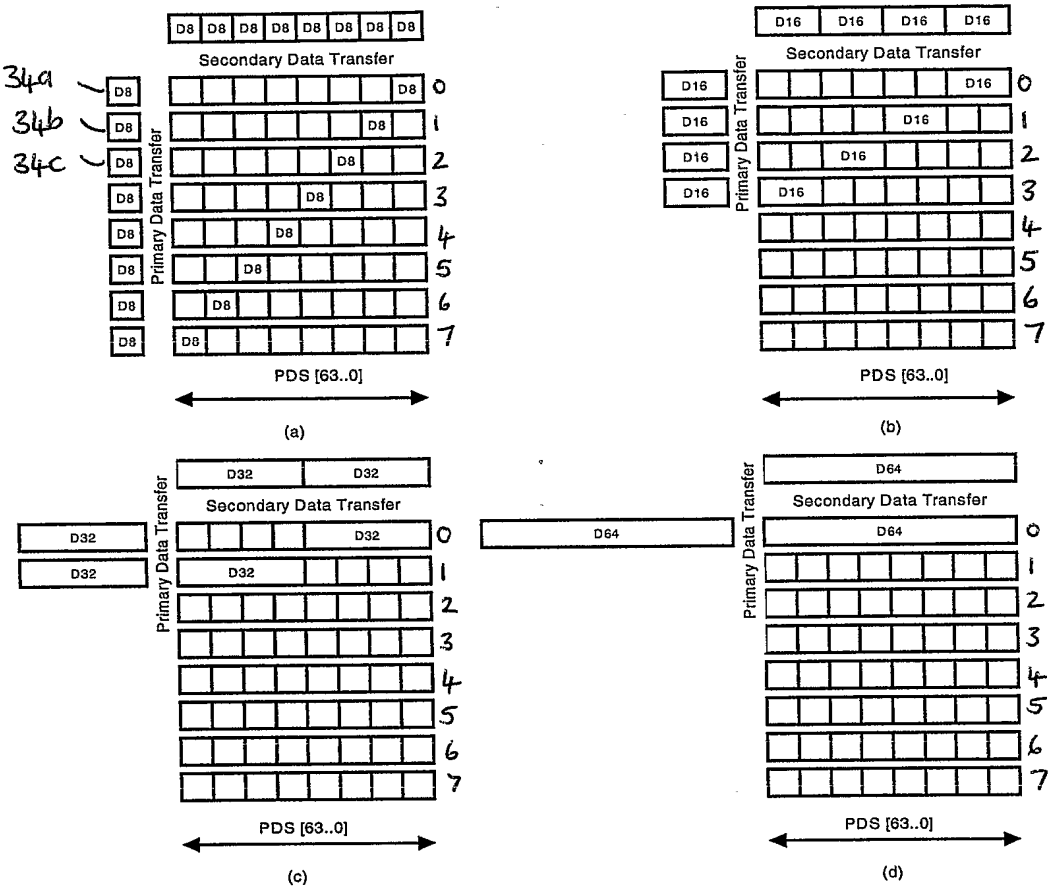

D8 D8 D8 D8 D8 D8 D8 D8
Secondary Data Transfer
34a
34b
34c
D8
Primary Data Transfer
PDS [63..0]
(a)
D16 D16 D16 D16
Secondary Data Transfer
D16
Primary Data Transfer
PDS [63..0]
(b)
D32 D32
Secondary Data Transfer
D32
Primary Data Transfer
PDS [63..0]
(c)
D64
Secondary Data Transfer
D64
Primary Data Transfer
PDS [63..0]
(d)

Fig. 7

byte address	column address	Row Strobe logic conditions
0	7..0	PDTEN _A[7..0] = BS[0].CS[7..0]
		PDTEN _B[7..0] = ~D8. BS[0].CS[7..0]
		PDTEN _C[7..0] = (D32+D64). BS[0].CS[7..0]
		PDTEN _D[7..0] = D64. BS[0].CS[7..0]
1	15..8	PDTEN _A[15..8] = (D8+D64). BS[1].CS[15..8]
		PDTEN _B[15..8] = ~D8. BS[1].CS[15..8]
		PDTEN _C[15..8] = (D32+D64). BS[1].CS[15..8]
		PDTEN _D[15..8] = D64. BS[1].CS[15..8]
2	24..16	PDTEN _A[24..16] = ~D16. BS[2].CS [24..16]
		PDTEN _B[24..16] = (D16+D64). BS[2].CS [24..16]
		PDTEN _C[24..16] = (D32+D64). BS[2].CS [24..16]
		PDTEN _D[24..16] = D64. BS[2].CS [24..16]
3	31..25	PDTEN _A[31..25] = (D8+D64). BS[3].CS [31..25]
		PDTEN _B[31..25] = ~D8. BS[3].CS [31..25]
		PDTEN _C[31..25] = (D32+D64). BS[3].CS [31..25]
		PDTEN _D[31..25] = D64. BS[3].CS [31..25]
4	39..32	PDTEN _A[31..25] = (D8+D64). BS[4].CS [39..32]
		PDTEN _B[31..25] = ~D8. BS[3].CS [39..32]
		PDTEN _C[31..25] = (D32+D64). BS[4].CS [39..32]
		PDTEN _D[31..25] = D64. BS[4].CS [39..32]
5	47..40	PDTEN _A[24..16] = ~D16. BS[5].CS [47..40]
		PDTEN _B[24..16] = (D16+D64). BS[5].CS [47..40]
		PDTEN _C[24..16] = (D32+D64). BS[5].CS [47..40]
		PDTEN _D[24..16] = D64. BS[5].CS [47..40]
6	55..48	PDTEN _A[15..8] = (D8+D64). BS[6].CS[55..48]
		PDTEN _B[15..8] = ~D8. BS[6].CS[55..48]
		PDTEN _C[15..8] = (D32+D64). BS[6].CS[55..48]
		PDTEN _D[15..8] = D64. BS[6].CS[55..48]
7	63..56	PDTEN _A[7..0] = BS[7].CS[63..56]
		PDTEN _B[7..0] = ~D8. BS[7].CS[63..56]
		PDTEN _C[7..0] = (D32+D64). BS[7].CS[63..56]
		PDTEN _D[7..0] = D64. BS[7].CS[63..56]

Table 4

Fig. 8

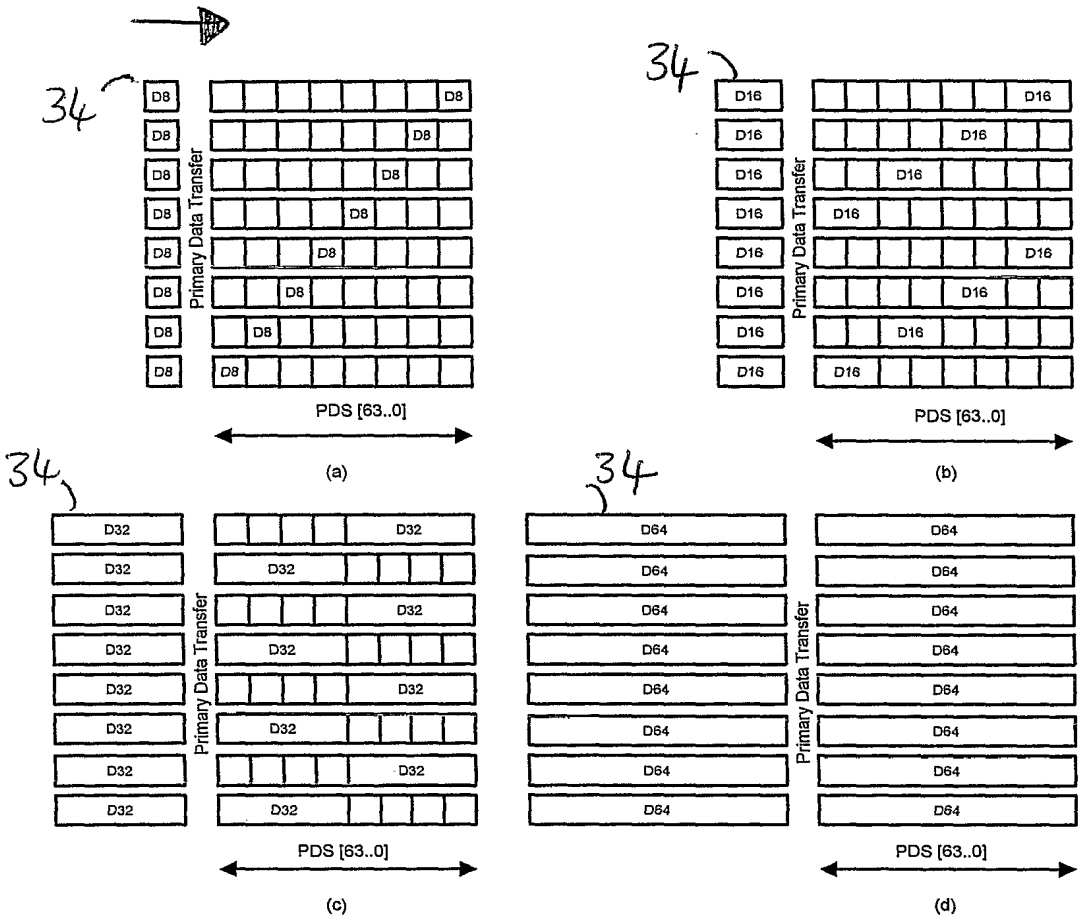

34
D8
Primary Data Transfer
PDS [63..0]
(a)
34
D16
Primary Data Transfer
PDS [63..0]
(b)
34
D32
Primary Data Transfer
PDS [63..0]
(c)
34
D64
Primary Data Transfer
PDS [63..0]
(d)

Fig. 9